MATRIX REPRESENTATION IN GRAPH THEORY

Dissertation submitted in partial fulfillment of the requirements for the

BACHELOR'S DEGREE IN MATHEMATICS

Jointly By

Annmary M M - 210021031513

Salini K S - 210021031538

Vignesh Muraleedhar – 210021031541

Under the supervision and guidance of

Mr. Toby B Antony

Assistant Professor

Department of Mathematics

Bharata Mata College, Thrikkakara

2021 - 2024

DECLARATION

We hereby declare that this project report entitled "MATRIX REPRESENTATION IN GRAPH THEORY" is a bonafide record of work done by us under the supervision of Mr. Toby B Antony and the work has not previously formed the basis for the award of any academic qualification fellowship or another similar title of any other university or board.

Annmary M M :

Salini K S :

Vignesh Muraleedhar :

Place: Thrikkakara

Date: 23.04.2024

CERTIFICATE

This is to certify that dissertation entitled “MATRIX REPRESENTATION IN GRAPH THEORY” submitted jointly by Miss Annmary M M, Miss Salini K S, Mr Vignesh Muraleedhar, in partial fulfillment of the requirements for the B.Sc. Degree in Mathematics is a bonafide record of the studies undertaken by them under my supervision at the Department of Mathematics, Bharata Mata College, Thrikkakara during the academic year 2021-2024. This dissertation has not been submitted for any other degree elsewhere.

Place: Thrikkakara

Date: 23.04.2024

Toby B Antony

(Supervisor)

ACKNOWLEDGEMENT

We have immense pleasure in presenting this report on "MATRIX REPRESENTATION IN GRAPH THEORY". We would like to express our sincere thanks to Mr. Toby B Antony, Assistant Professor of the Department of Mathematics, Bharatha Mata college, Thrikkakara, for his valuable support and guidance that enabled us to make study in this topic and to prepare this dissertation.

We are also thankful to all other teachers in the department for their cooperation to prepare this report. We also extend our sincere thanks to the staff and all our friends for their help and support during this study.

Annmary M M

Salini K S

Vignesh Muraleedhar

Place: Thrikkakara

Date: 23.04.2024

CONTENTS

ABSTRACT

Chapter 1 includes the basics of graphs and matrices.

Chapter 2 discuss about path matrix.

Chapter 3 discuss about cut set matrix.

Chapter 4 discuss about degree matrix

Chapter 5 discuss about Laplacian matrix.

Chapter 6 contains the applications of matrix representation in graph theory.

INTRODUCTION

Graph Theory is a branch of Mathematics that studies the relationships between objects represented as vectors and the connections between these objects represented as edges. It finds applications in various fields, including computer science, biology, social science, and network analysis.

The Königsberg bridge problem, famously solved by Leonhard Euler in1736, is often considered as the starting point of graph theory. Euler's solution laid the groundwork for the development of Graph theory. Euler addressed the problem of whether it was possible to traverse the seven bridges of Königsberg exactly once and return to the starting point without retracing any paths. Euler's solution introduced the concept of vertices, edges and paths, forming the basis of modern graph theory.

One of the key aspects of graph theory is the representation of graphs using matrices, providing a powerful and efficient framework for analyzing graph structures. there are several matrices which can be associated with graphs. In this study we discuss about the matrix representation in graph theory, exploring its basic concepts, application and implication in various fields. From adjacency matrix to Laplacian matrix, each representation provides unique insight into underlying structure of the graph, allowing us to unravel complex networks and phenomena.

CHAPTER 1
BASIC CONCEPTS

1.1 GRAPHS

A *graph* G is a triple consisting of a vertex set $V(G)$, an edge set $E(G)$, and a relation that associates with each edge two vertices (not necessarily distinct) called its end vertices.

The *order* of a graph, denoted by $|V|$, refers to the number of vertices it contains and the *size* of a graph, denoted as $|E|$, refers to the number of edges it contains.

Directed Graphs: *Directed graph* is a graph where edges have directions.

Undirected Graph: *Undirected graph* is a graph where edges have no directions.

Incidence: When a vertex v_i is an end vertex of some edge e_i then v_i and $e_i$are said to be incident to each other.

Loop: A *loop* is an edge that connects a vertex to itself.

Degree: Number of edges incident on a vertex v is called degree of v. It is denoted by $d(v)$.

Pendant Vertex: A vertex v of G is called a *pendant vertex* if and only if v has degree 1.

Isolated Vertex: An *isolated vertex* in a graph is a vertex that has degree 0.

End Vertex: An *end vertex* refers to a vertex in a graph that has only one incident edge.

Walk: A *walk* in a graph is a finite sequence W: $v_0, e_1, v_1, e_2, v_2, e_3, v_3, \dots, e_k, v_k$ whose elements are alternately vertices and edges.

Path: If the vertices of the walk W are distinct then W is called a *path.*

Length of a path: The *length of a path* is the number of edges it contains.

Trail: If the edges $e_1, e_2, e_3, \dots, e_k$ of the walk W: $v_0, e_1, v_1, e_2, v_2, e_3, v_3, \dots, e_k, v_k$ are distinct, then W is called a *trial.*

Cycle: A non-trivial closed trail in a graph is G is called a *cycle*

Simple Graph: A graph which does not have any loop or parallel edges is called *simple graph.*

Complete Graph: A *complete graph* is a simple graph in which each pair of distinct vertices is joined by an edge. A complete graph of order n is denoted as K_n.

Connected Graph and Disconnected Graph: If there is a path between every pair of vertices is called *connected graph.* If it is it not connected then it is called *disconnected.*

Weighted Graph: A *weighted graph* is a type of graph where each edge has an assigned a numerical value called a weight. These weights represent various quantities such as distance, costs or capacities

Subgraph: A graph H is called a *subgraph* of a graph G if $V(H)$ is a subset of $V(G), E(H)$ is a subset of $E(G)$.

Regular Graph: A graph is called a *regular graph* if the degree of each vertex is equal.

Tree: A *tree* is a connected graph that has no cycles.

First theorem of graph theory: It states that the sum of degree of vertices of a graph G is twice the number of edges of G.

1.2 MATRIX

A *matrix* is a rectangular array of numbers, symbols, or expressions, arranged in rows and columns.

Adjacency Matrix: It represents the connections between vertices in a graph.

Incidence Matrix: Represents the relationships between vertices and edges.

Rank of a matrix: The maximum number of linearly independent rows or columns of a matrix is called the *rank* of a matrix.

Determinant: It is a scalar value that can be computed from the elements of a square matrix.

Trace: The *trace* of a matrix A is defined to be the sum of elements on the main diagonal A. It is only defined for $n \times n$ matrices.

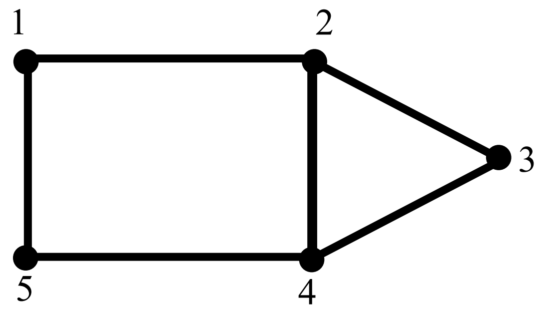
Degree sum of a matrix: The *degree sum of a matrix* refers to the sum of the degrees of its vertices in a graph representation.

1.3 ADJACENCY MATRIX

The *adjacency matrix* of G with n vertices and no parallel edges is an $n \times n$ symmetric binary matrix $A = [\, a_{ij}]$,

$$a_{ij} = \begin{cases} 1 \text{ if there is an edge between } v_i \text{ and } v_j \\ 0 \text{ if there is no edge between } v_i \text{ and } v_j \end{cases}$$

Example: Consider the graph below



Its corresponding adjacency matrix is'

$$A(G) = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

We have the following observations about adjacency matrix:

- The diagonal entries of an adjacency matrix are zero if the graph has no loops.
- If the graph G is an undirected graph, then the adjacency matrix $A(G)$ is symmetric.
- A row with single unit entry corresponds to a pendent vertex and the row with all entries zero corresponds to an isolated vertex.

POWER

If A is a square matrix and k is a positive integer, the k^{th} power of A is given by $A = A \times A \times \cdots \times A$, where A is multiplied k times.

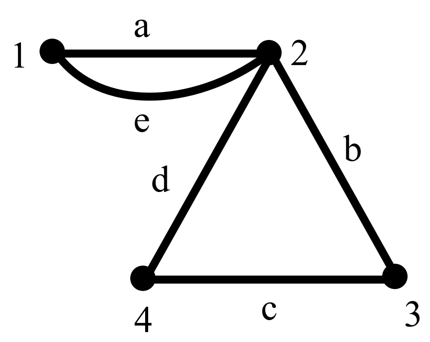
In the matrix A^k, where A is an adjacency matrix of a graph with vertices $v_1, v_2,, v_3, \dots v_n$, the elements (i, j) give the number of walks of length k from v_i to v_j.

1.4 INCIDENCE MATRIX

Let G be a graph with n vertices $v_1, v_2,, v_3, \dots v_n$, m edges$e_1, e_2,, e_3, \dots e_m$ and no self-loops. Then the *incidence matrix* is an $n \times m$ matrix $M = [m_{ij}]$, whose n rows correspond to n vertices and the m columns correspond to the m edges, as follows:

$$m_{ij} = \begin{cases} 1, \text{ if } v_i \text{ is an end vertex of } e_j \\ 0, \text{ if } v_i \text{ is not an end vertex of } e_j. \end{cases}$$

Example: Let G be a graph as given below.



The Incidence matrix is given by,

$$M(G) = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

We have the following observations about incidence matrix:

- Each column entry contains exactly two unit entry.
- Parallel edges in a graph produces identical columns in its incidence matrix.
- Each vertex v_i of $V(G)$ has a degree equal to the number of unit entries corresponding to v_i.
- A row with single unit entry corresponds to a pendent vertex, and the row with all entries is zero corresponds to an isolated vertex.
- If G is a graph with incidence matrix $M(G)$, then the column sum of $M(G)$ is zero and row sum will give you the rank.

CHAPTER 2
PATH MATRIX

2.1 PATH MATRIX

In graph theory, a path matrix is a matrix representation that shows the links or paths between vertices in a graph. If a path exists between two vertices in the graph, it is indicated by each element in the matrix. It's frequently used in algorithms that look for the shortest paths or examine a graph's connection.

DEFINITION

If P is a path in G, then the path matrix is defined for vertices say (x, y) as

$$P\ (x, y) = [p_{ij}] \text{ where,}$$

$$P_{ij} = \begin{cases} 1 \text{ if } j^{th} \text{ edge lies in the } i^{th} \text{ path} \\ 0 \text{ otherwise} \end{cases}$$

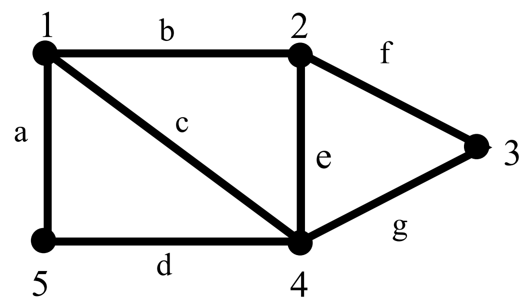
.

CONSTRUCTION

The construction of path matrix is as follows:

1) Consider a graph G with vertex set $V(G)$ and edge set $E(G)$.Take two vertices, say x and y from $V(G)$.
2) Determine all the paths between x to y.
3) In the Path matrix, the rows correspond to the different paths between x and y and the columns corresponds to the edges in G.
4) If the j^{th} edge lies in the i^{th} path, then p_{ij} will be one and zero otherwise.

Example: Consider the given graph.



The paths from 1 to 4 is given by

$P_1 = 1 - a - 5 - d - 4$,

$P_2 = 1 - b - 2 - e - 4$,

$P_3 = 1 - b - 2 - f - 3 - g - 4$,

$P_4 = 1 - c - 4$

	a	b	c	d	e	f	g
P_1	1	0	0	1	0	0	0
P_2	0	1	0	0	1	0	0
P_3	0	1	0	0	0	1	1
P_4	0	0	1	0	0	0	0

Its path matrix is given by,

$$P(1,4) = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

We have the following observations about path matrix:

- A column with all 0's corresponds to an edge that does not lie in any path between x and y.
- A column with all 1's corresponds to an edge that lies in every path between x and y.
- There is no row with all 0's.

PROPERTIES

1) If there is a path from vertex v to vertex u and a path from vertex u to vertex w, then there is a path from vertex v to vertex w. This property is reflected in the path matrix by the presence of indirect paths between vertices.
2) The path matrix can be used to determine the connectivity of a graph. If all the path matrices of a graph are non-zero, then every pair of vertices in the graph is connected by at least one path and hence the graph is connected. If any path matrix of the graph is a zero matrix, then the graph is disconnected.
3) By examining the entries of the path matrix, one can determine the shortest path lengths between any pair of vertices in the graph. The row with least number of 1's represents the shortest path whereas a row with highest number of 1's represents the longest path

Theorem:

Shortest Path Theorem: In a weighted graph, the shortest path between two vertices is the one with the minimum sum of weights.

Proof: Define a weighted graph $G = (V, E)$ with vertex set $V(G)$ and edge set $E(G)$, where each edge has a weight associated with it. Claim that P is the shortest path between vertices u and v in G.

Assume there exists another path P' between u and v with a shorter length. Then if P' is shorter than P, the sum of weights along P' is less than the sum of weights along P. Since P' is a path, it must consist subpaths. By replacing the subpath in P with the corresponding subpath in P', we create a new path P''. Since P'' contains parts of P and P', the sum of weights along P'' is less than the sum of weights along P.This is a contradiction to our assumption that P is the shortest path. Therefore, P must be the shortest path between u and v.

CHAPTER 3
CUT SET MATRIX

In graph theory, cut set matrices are used in various graph theoretic problems, such as network connectivity, graph partitioning, and network analysis. A cut set matrix is a binary matrix that captures the connectivity of a graph. It helps in understanding the structure of a graph and identifying certain edges whose removal can affect its connectivity.

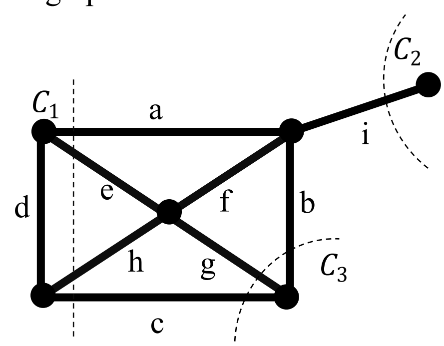
CUT SET

A *cut* is a method of dividing the graph into two or more separate group of vertices, so that removing the edges connecting these groups causes the graph to be disconnected.

In a connected graph G, a *cut set* is a set of edges whose removal from G leaves the graph disconnected, provided removal of no proper subset of these edges disconnects G.If there are n cut sets, it can be denoted as $C_1, C_2, \ldots, C_n$.

Cut sets are primarily defined in terms of edges rather than vertices in graph theory. While cut sets are defined for edges, vertex cuts are defined for vertices, which refers to a subset of vertices in a graph such that removing these vertices disconnects the graph into two or more disjoint components.

Example: Let G be the given graph



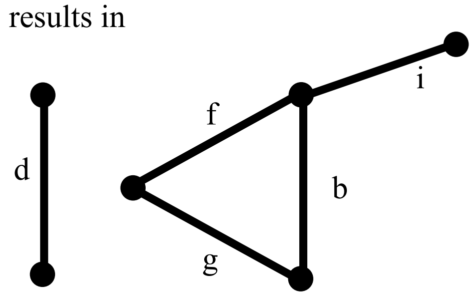
In this graph, some of the cut sets are,

C_1= {a, e, h, c}

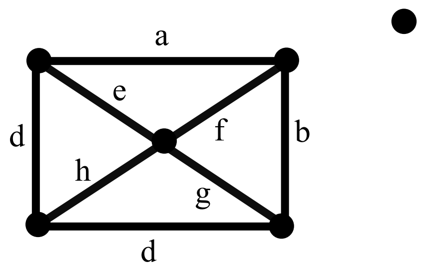
C_2= {i}

C_3= {c, g, b}

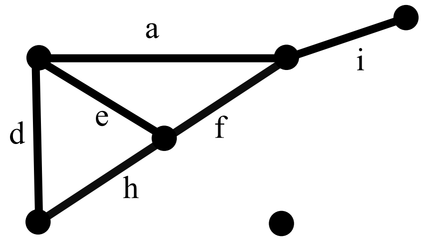
Removal of C_1 results in



Removal of C_2 results in



Removal of C_3 results in



Here, the set {a, i} is not a cut set. By definition of cut set, no proper subset of the considered set should disconnect the graph. In the set {a, i}, the proper subset {i} disconnects the graph.

CUT SET MATRIX

DEFINITION

Let G be a graph. Then the *cut-set matrix* $C = [c_{ij}]$ is defined to be an $m \times n$ matrix, where each row corresponds to the cut sets in the graph, and each column corresponds to the edge of the graph.

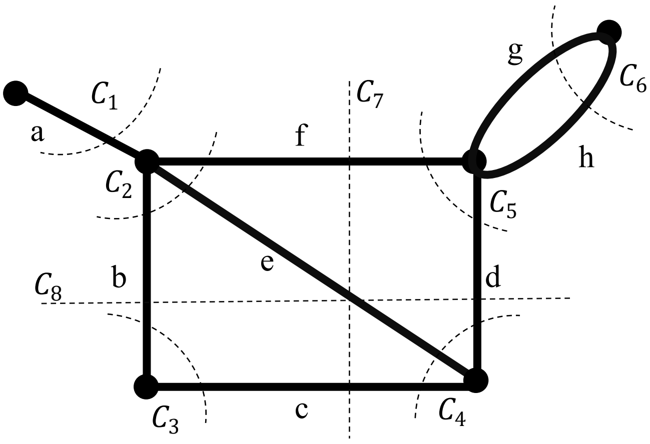
$$c_{ij} = \begin{cases} 1; \text{ if } i^{th} \text{ cut set contains } j^{th} \text{ edge} \\ 0; \text{ otherwise} \end{cases}$$

CONSTRUCTION

For a graph G the cut set matrix is created as follows:

1) Determine all the possible cut sets in the given graph, i.e. find the set of edges whose removal will make the graph disconnected.
2) Then we have a matrix with dimension $m \times n$, where m is the number of cut sets in the graph and n is the number edges in the graph.
3) Set c_{ij} to 1 if i^{th} cut set contains j^{th} edge. If not, set c_{ij} to 0.

Example: Consider the given graph G,



The cut sets of the above graph are

C_1= {a}

C_2= {b, e, f}

C_3= {b, c}

C_4= {c, e, d}

C_5= {f, d}

C_6= {g, h}

C_7= {f, e, c}

C_8= {b, e, d}

	a	b	c	d	e	f	g	h
C_1	1	0	0	0	0	0	0	0
C_2	0	1	0	0	1	1	0	0
C_3	0	1	1	0	0	0	0	0
C_4	0	0	1	1	1	0	0	0
C_5	0	0	0	1	0	1	0	0
C_6	0	0	0	0	0	0	1	1
C_7	0	0	1	0	1	1	0	0
C_8	0	1	0	1	1	0	0	0

Then the corresponding cut set matrix is given by,

$$C(G) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

PROPERTIES

1) The cut set matrix is a binary matrix, meaning its entries are either 0 or 1.
2) The cut set matrix helps to identify the minimum set of edges whose removal disconnects the graph. The row with least number of 1's gives the minimum cut set.
3) A 1 in a row of the cut set matrix indicates that removing the edges associated with that row would disconnect the graph or increase its number of connected components.
4) The cut set matrix provides information about the graph's connectivity.
5) The cut set matrix is often sparse, meaning that it contains mostly zero entries.
6) The entries in the cut set matrix that are zero indicates that removing those edges does not affect the graph's connectivity.

CHAPTER 4
DEGREE MATRIX

In graph theory, the degree matrix is a square matrix that shows the degree of each vertex in a graph. The degree matrix is a fundamental component in various graph algorithms providing essential information about graph connectivity and structure of graph. It is often denoted as D and is derived from adjacency matrix.

DEFINITIONS

Given an undirected graph G with n vertices, the *degree matrix* $D = [d_{ij}]$ is an $n \times n$ diagonal matrix, where each diagonal element d_{ii} represent the degree of vertex i.

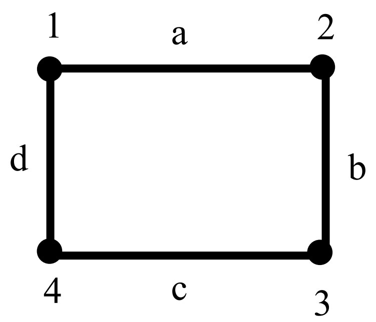
$$d_{ij} = \begin{cases} \deg(v_i), & \text{if } i = j \\ 0, & \text{if } i \neq j \end{cases}$$

CONSTRUCTION

For a given graph G, the degree matrix D is created as follows:

1) Start a matrix with dimensions of $n \times n$, where n is the number of vertices in G.
2) For each vertex i in G
    - Let d_i be the degree of the vertex i, or the number of edges incident to vertex i..
    - Assign the i^{th} diagonal element of D to d_i.
    - In the i^{th} row and i^{th} column, set the values of all other elements to zero.
3) Repeat step 2 for all the vertices in G to finish the construction of D.

Example: Consider a simple graph G with 4 vertices



Its corresponding degree matrix is given below:

$$D(G) = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}$$

Here each vertex has a degree of 2, so the diagonal elements are 2.

We have the following observations about degree matrix:

- The diagonal entries represent the degrees of individual vertices and all other entries will be zero.
- The total number of edges can be found by dividing the sum of all the diagonal elements in the degree matrix by 2.
- In an undirected graph, if the diagonal entries are non-zero, then every vertex is connected to at least one other vertex.

PROPERTIES

1) The degree matrix $D(G)$ is symmetric.
2) The degree matrix is a diagonal matrix, meaning all of its off-diagonal elements are zero. The reason for this characteristic is that the degree of the vertex influences just its own entry in the degree matrix.
3) The entries in the degree matrix are non-negative integers since the vertex's degree specifies the number of edges that pass through it, which cannot be negative.
4) The degree matrix can be used to determine the connectivity of graph.
5) The degree matrix is a fundamental component in the construction of the Laplacian matrix of a graph. The Laplacian matrix is obtained by subtracting the adjacency matrix from a scaled version of degree matrix.
6) Connection with adjacency matrix

The connection between the degree matrix and the adjacency matrix can be established through the adjacency matrix itself.

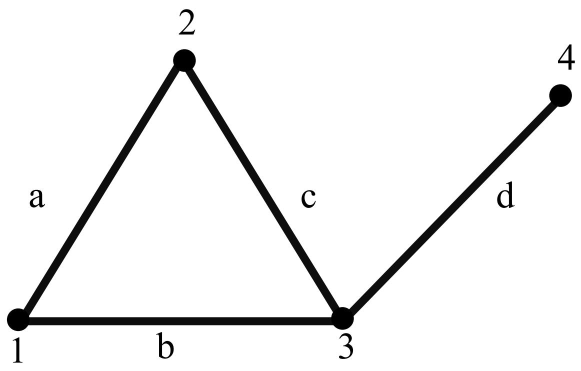
Let A be an adjacency matrix of a graph with n vertices.

Then the degree matrix $D = [d_{ij]}$ can be defined using the adjacency matrix A as follows:

$$d_{ij} = \begin{cases} \sum_{j=1}^{n} a_{ij}, & \text{if } i = j \\ 0, & \text{if } i \neq j \end{cases}$$

The i^{th} diagonal entry in the degree matrix is the sum of all entries in the i^{th} row of the adjacency matrix. This connection is because each entry in the i^{th} row of the adjacency matrix corresponds to an edge incident to vertex v_i, and the sum of these entries give the degree of v_i.

Example: Let G be a graph with $V(G) = \{1,2,3,4\}$ and $E(G) = \{a, b, c, d\}$ as given below



Its adjacency matrix $A(G)$ is given by

$$A(G) = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

The degree matrix of graph G is given by

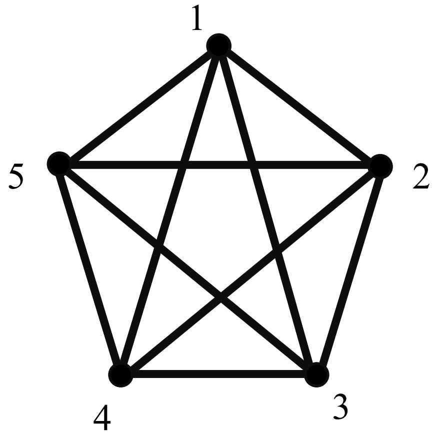
$$D(G) = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Here, the sum of entries of the first row of $A(G)$ is equal to the first diagonal entry in $D(G)$.

In general, k^{th} row sum of $A(G)$ is equal to the k^{th} diagonal entry in D(G).

7) The degree matrix of a k-regular graph is a diagonal matrix with all diagonal entries as k.

Example: Consider a 4-regular graph as given below

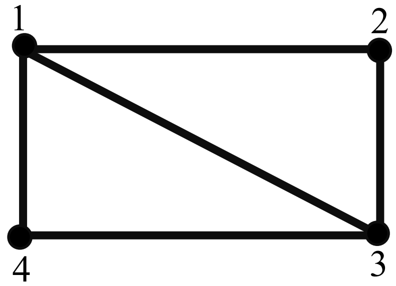


The degree matrix is given by,

$$D = \begin{bmatrix} 4 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 & 4 \end{bmatrix}$$

8) The trace of a degree matrix will be equal to $2n$, where n is the number of edges in the considered graph.

Example: Let G be a graph as given below



Here the degree matrix is,

$$D = \begin{bmatrix} 3 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}$$

Trace of D = 3+2+3+2 = 10

Here, number of edges = 5. Therefore, $2n$ = 2×5 = 10

Hence, $2n$ = Trace of D

CHAPTER 5
LAPLACIAN MATRIX

Laplacian Matrix is a fundamental concept in graph theory, serving as the mathematical representation of the connectivity and structure of graph. It is often denoted as L and can be derived from the Adjacency Matrix A of an undirected graph.

DEFINITION

For an undirected graph with n vertices, the Laplacian matrix $L = [l_{ij}]$ is defined as the difference between the degree matrix D and the adjacency matrix. A.

$$L = D - A$$

where D is the degree matrix and A is the adjacency matrix.

The Laplacian Matrix can be defined as follows:

$$l_{ij} = \begin{cases} \text{Deg } (v_i);\ \text{if } i = j \\ \text{-1; if } i \neq j \text{ and } v_i \text{ is adjacent to } v_j \\ \text{0; otherwise} \end{cases}$$

CONSTRUCTION

A graph's degree matrix and adjacency matrix are used to build the graph's Laplacian Matrix. The construction of a graph's Laplacian Matrix L is as follows:

1) Degree matrix (D)
   - Create a diagonal matrix D with n rows and n columns, where n is the number of vertices in the graph.
   - The diagonal elements of D represent the degrees of vertices in the graph.
   - For each vertex i of degree d_i, set the i^{th} diagonal element of D to d_i and all other elements to zero.

2)

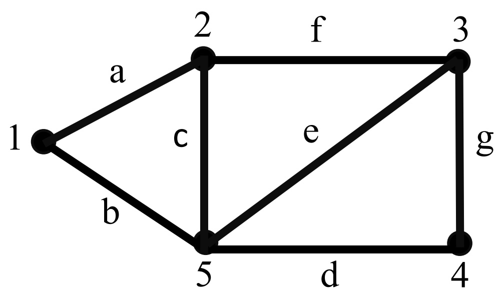
3) Adjacency matrix (A)

- Create the graph's adjacency matrix A.
- Set a_{ij} and a_{ji} to 1 if the vertices i and j are nearby. If not, make them equal to 0.
- The diagonal elements of A stand for loops.

4) Laplacian matrix (L)

- To obtain the Laplacian matrix L, subtract the degree matrix D from the adjacency matrix A.
- Every element l_{ij} of L is calculated as $d_{ij} - a_{ij}$.
- Alternatively, $L = D - A$ can be expressed as a matrix subtraction operation.

Example: Consider the graph G with vertex set $V(G) = \{1,2,3,4,5\}$ and edge set $E(G) = \{a, b, c, d, e, f, g\}$ as given below



Its corresponding adjacency matrix and degree matrix are given by

$$A(G) = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

$$D(G) = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 4 \end{bmatrix}$$

Hence, the Laplacian matrix is

$$L(G) = \begin{bmatrix} 2 & -1 & 0 & 0 & -1 \\ -1 & 3 & -1 & 0 & -1 \\ 0 & -1 & 3 & -1 & -1 \\ 0 & 0 & -1 & 2 & -1 \\ -1 & -1 & -1 & -1 & 4 \end{bmatrix}$$

PROPERTIES

1) The Laplacian matrix $L(G)$ is symmetric.
2) Every row sum and column sum of a $L(G)$ is zero.
3) The trace of the $L(G)$ will be equal to $2m$, where m is the number of edges in the considered graph.
4) The rank of $L(G)$ is $n-k$, where k is the number of connected components of G. If G is connected, then the rank of $L(G)$ is $n-1$.
5) The off-diagonal entries of a $L(G)$ are nonpositive.
6) Connection with incidence matrix

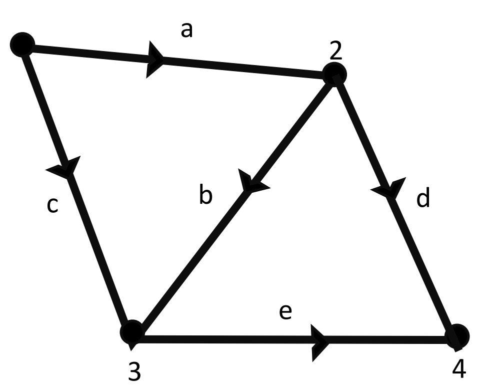
Suppose G is a graph in which each edge is assigned an orientation. The incidence matrix $M(G) = m_{ij}$ is the $m \times n$ matrix defined as follows. The $(i,j)^{th}$ entry of $M(G)$ is 0 if vertex v_i and edge e_j are not incident and otherwise it is 1 or -1 according as e_j originates or terminates at i respectively.

$$L(G) = M(G)M(G)^T$$

$M(G)^T$ is the transpose of $M(G)$.

Example:

Let G be a graph with $V(G) = \{1, 2, 3, 4\}$ and $E(G) = \{a, b, c, d, e\}$ as given below



Then, $M(G) = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ -1 & 1 & 0 & 1 & 0 \\ 0 & -1 & -1 & 0 & 1 \\ 0 & 0 & 0 & -1 & -1 \end{bmatrix}$

$$M(G)^T = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

Its Laplacian matrix is given by

$$L(G) = \begin{bmatrix} 2 & -1 & -1 & 0 \\ -1 & 3 & -1 & -1 \\ -1 & -1 & 3 & -1 \\ 0 & -1 & -1 & 2 \end{bmatrix}$$

$$M(G)M(G)^T = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ -1 & 1 & 0 & 1 & 0 \\ 0 & -1 & -1 & 0 & 1 \\ 0 & 0 & 0 & -1 & -1 \end{bmatrix} \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

$$M(G)M(G)^T = \begin{bmatrix} 2 & -1 & -1 & 0 \\ -1 & 3 & -1 & -1 \\ -1 & -1 & 3 & -1 \\ 0 & -1 & -1 & 2 \end{bmatrix} = L(G)$$

CHAPTER 5

APPLICATIONS

1) Network Analysis

Graphs are used to model networks in various domains such as social networks, communications networks, and transportation networks. Matrix representations help analyze connectivity, flow, and centrality in these networks.

The adjacency matrix A encodes the connections between nodes in the network. In network analysis, adjacency matrices are used to represent network topology and study various properties such as connectivity, paths, and cycles. By performing matrix operations on adjacency matrix, network analysts can determine node degrees, clustering coefficients. The Laplacian matrix is used to study network dynamics, diffusion process and synchronization phenomena. Degree matrix represents the degrees of the nodes in the network. It is used to calculate various node-level and network-level metrics. By analyzing degree matrix, network analysts can identify hubs, and critical nodes in the network and evaluate its resilience to node failures or attacks.

2) Computer Science and Algorithms

Matrix representations are fundamental in analyzing graph algorithms. The adjacency matrix, for example, is used in algorithms for traversing graphs, finding connected components, and detecting cycles.

The adjacency matrix is widely used in graph traversal algorithms, shortest path finding and connectivity analysis. For example, breadth-first search (BFS) and depth-first search (DFS) algorithms can be implemented efficiently using adjacency matrix. Laplacian matrix captures important structural properties of a graph and is used in graph partitioning, clustering, and spectral analysis algorithms. Degree matrix is used in algorithms for centrality analysis, network flow optimization.

3) Spectral Graph Theory

Spectral properties of matrices associated with graphs, such as Laplacian matrix, have applications in clustering, partitioning, and understanding the global structure of networks

The Laplacian matrix is a key matrix in spectral graph theory. It captures important structural properties of a graph, including its connectivity, clustering, and spectral characteristics. Spectral graph theory uses the eigenvalues and eigenvectors of the Laplacian matrix to study various properties of graphs, such as clustering, partitioning and to understand complex network structures.

4) Optimization Problems

Matrix representations are used to formulate and solve optimization problems related to graphs, such as finding the minimum spanning tree, shortest paths, and maximum flow in networks.

In optimization problems such as travelling salesman problem (TSP), adjacency matrix is used to encode costs or weights associated with edges. By analyzing the adjacency matrix, optimization algorithms can determine shortest path, or minimum spanning tree. Incidence matrix is used to model resource flows or constraints in graphs. By analyzing the columns of incidence matrix, optimization algorithms can determine the optimal resource flow, or satisfy linear constraints. Also, by analyzing the entries of degree matrix, optimization algorithms can identify important nodes, or maximize network centrality.

5) Game Theory

Graphs are used to model strategic interactions in game theory. Matrix representation in graph theory provides a powerful framework for analyzing strategic interactions and modelling games played on a network. By taking advantage of matrix representations and related techniques, game theorists can study the dynamics of strategic interactions, determine equilibrium outcomes, and analyze the effects of structure of network on game outcome.

CONLUSION

To summarize, matrix representation is important in graph theory because it provides a powerful framework for analyzing and comprehending graph features. Graph can be analyzed from an algebraic and combinatorial perspective using several matrix representations such as the adjacency matrix, the incidence matrix, Laplacian matrix and degree matrix.

Matrix representation is a fundamental concept in graph theory, allowing for the investigation, modelling, and solution of a wide range of real-world issues across various domains.

REFERENCE

1) R. Balakrishnan and K. Ranganathan; A Text Book of Graph Theory.

2) Frank Hararay; Graph theory

3) John Clark and Cerek Allan Holtan; A First Look at Graph Theory.

4) R. B. Bapat; Graphs and Matrices.